Beware of the Bias – Statistical Performance Evaluation of Higher-Order Alphabet PUFs

Christoph Frisch, Michael Pehl
Department of Electrical and Computer Engineering
Technical University of Munich
Munich, Germany
{chris.frisch, m.pehl}@tum.de

***Abstract*—Physical Unclonable Functions (PUFs) derive unpredictable and device-specific responses from uncontrollable manufacturing variations. While most of the PUFs provide only one response bit per PUF cell, deriving more bits such as a symbol from a higher-order alphabet would make PUF designs more efficient. This type of PUFs is thus suggested in some applications and subject to current research. However, only few methods are available to analyze the statistical performance of such higher-order alphabet PUFs. This work, therefore, introduces various novel schemes. Unlike previous works, the new approaches involve statistical hypothesis testing. This facilitates more refined and statistically significant statements about the PUF regarding bias effects. We utilize real-world PUF data to illustrate the capabilities of the tests. In comparison to state-of-the-art approaches, our methods indeed capture more aspects of bias. Overall, this work is a step towards an improved quality control of higher-order alphabet PUFs.**

***Index Terms*—PUF, quality evaluation, bias, metrics**

I. Introduction

The demand for security even in low-cost devices accelerated the rise of a new security primitive: Physical Unclonable Functions (PUFs). Through their ability to derive device-unique responses from the unavoidable minuscule variations of the design process, they open the door for cheap key storage, new authentication protocols, and tamper-resistance. Most PUFs, such as ring oscillator (RO), SRAM or Arbiter PUFs, measure the variations and quantize them to a single bit per measurement circuit and challenge. However, quantization to a higher-order alphabet would be beneficial: When realized in a sensible way, it allows for deriving more secret bits from the same area, i.e. the entropy per area is increased. Therefore, this idea has been researched in several approaches until now. While optical PUFs – which provide, e.g., a speckle pattern and might be accounted to the class of PUFs outputting symbols from an higher-order alphabet – are out of the scope of this work, already the Lightweight Secure PUF [1] describes a primitive outputting multiple bits from a single PUF instance. Recent research in the domain of RO PUFs tries to improve efficiency by quantizing the analog measurement result into a multibit value [2]. Another approach is described in [3]: A secret is derived from measuring capacitors in a mesh of wires used to protect the underlying system against tampering and to derive a secret key. The capacitors are quantized to responses in a higher-order alphabet, in order to reach sufficient sensitivity against tampering and to extract a high amount of entropy.

Because PUF responses are used to derive a device-unique secret, for any PUF the unpredictability of these responses is an important criterion. Several works to evaluate this property are outlined in Section II. Most have in common that they focus on PUFs with only two output symbols. In addition, only little effort is spent on evaluating PUFs regarding statistical significance and ensuring confidence in the results. But without confidence in the evaluation result, certification of PUFs will not be possible, which will become especially an issue for PUFs used for product protection.

Bias is one of the most important statistical properties of a PUF response. Bias means that a specific symbol in a response is more likely to appear than other symbols. Such defects are the easiest for an attack to observe and exploit. This motivates the focus of our work: We advance the evaluation of bias effects in Higher-order Alphabet PUFs (HoA-PUFs) with confidence and propose three tools to analyze a PUF's quality in that regard. Correlation effects are not considered in this work.

Contribution: Unlike for binary PUFs there hardly exist any metrics to evaluate the quality of HoA-PUFs. Consequently, this work proposes three methods to enable a thorough evaluation of bias effects. First, we explain these methods on a technical level and give several possibilities how they can assess a HoA-PUF's quality. A high quality thereby means that a HoA-PUF's bias is sufficiently small such that it can be mitigated by a suitable post-processing. Second, we apply them to exemplary real PUF data. This analysis also involves a comparison to state-of-the-art metrics. We can therefore illustrate the meaning of the results and highlight the variety of aspects that the new metrics cover.

Outline: In this paper, Section II presents the state of the art for PUF metrics as well as preliminary basics. Next, in Section III the new evaluation concepts are explained which we then apply to real PUF data in Section IV. Finally, we conclude the findings in Section V.

II. Preliminaries

A. State-of-the-art PUF Metrics

The predominant PUF metrics to assess the quality of a PUF are the metrics proposed in [4] and [5]. Together with tests from True Random Number Generator test suites, some of them have become part of a first standard for PUF evaluation [6]. The metrics in [4], [5] are quantitative approaches resulting

978-3-9819263-6-1/DATE22/©2022 EDAA

in a single value to compare PUFs to the ideal case or to one another. Although [5] additionally utilizes confidence intervals, the authors in [7] argue that there are drawbacks with the approach presented in [5]. Thus, they re-introduce the Bit-Alias now including confidence intervals and hypothesis testing. Correlation effects are evaluated in [8]. However, all of these metrics have been focusing on binary PUFs.

Some of the binary metrics were modified for HOA-PUFs [9] (e.g. Uniqueness and Reliability). Yet, fundamentally new approaches are required, because nearly all previous metrics for binary PUFs cannot be applied as such to HOA-PUFs. Compression has been used to evaluate HOA-PUFs entropy [10] but without evaluating the statistical significance of the results. In [2], the chi-squared test compares two approaches for quantizing normally distributed raw data to design a HOA-PUF. While related to one aspect of this work there are two main differences in how the chi-squared test is applied: First, the goal is not the same. In [2] the test was not proposed to analyze the quality of a PUF. Second, a comparison of χ^2 suffices in case of [2] and no hypothesis testing was utilized.

B. Quantization for HOA-PUFs

For HOA-PUFs such as in [2], [3] there is analog raw data which is drawn from a normal distribution. Two main strategies exist to derive symbols from a higher-order alphabet [11]: equidistant and equiprobable quantization. Equidistant quantization converts the raw data according to intervals of the same size. This is useful to support tamper-resistance [11]. However, symbols are not equally likely in this case so that the entropy per derived symbol is not optimal. Equiprobable quantization converts the analog data in a way that ensures that all symbols are equally likely. Thus, the entropy per derived symbol is higher than in the equidistant case. The equiprobable quantization is further improved in [2]. Since our focus is on key-extraction rather than on tamper-resistance and, thus, high entropy per symbol is crucial, we will discuss the novel evaluation schemes mostly with respect to equiprobable quantization. The general concepts, however, are also applicable to HOA-PUFs based on equidistant quantization.

Overall, we distinguish in the following between an *underlying* probability distribution and a *reference* probability distribution. The underlying distribution corresponds to the actual PUF. The samples have been drawn from the underlying distribution and we try to determine it based on these samples. The reference distribution refers to the expected symbol probabilities if we have a bias-free normal distribution of the raw data. It depends on the quantization and is used for a comparison with the underlying probability distribution.

C. Hypothesis Testing

The main tool for evaluating the quality of a HOA-PUF in this paper is statistical inference. This means we observe PUF responses as data and try to make statements about the underlying probability distribution. For example, we can test a statistical null hypothesis H_0. Although no statistical approach can actually prove something, we can at least make probability statements about an unknown statistic.

When testing a null hypothesis, there are four possible outcomes: (i) and (ii) we correctly do / do not reject the null hypothesis. (iii) we falsely reject a correct null hypothesis. This is a Type I error which occurs with probability α. (iv) We do not reject a false H_0. This is a Type II error with probability β. Overall, meaningful (statistically significant) statements are only possible by rejecting H_0. Not rejecting H_0 does not mean that H_0 is correct. Additionally, we cannot minimize both α and β. We therefore minimize β for a fixed α. In the following, we will state that α has to be chosen before the actual test.

III. Statistical Tests for Bias in HOA PUFs

When given the underlying probability distribution for each individual response symbol, it is easy to pinpoint any bias. However, in practice we do not know the probability distribution, but can only derive conclusions from samples drawn from the PUF or over multiple devices. This link between observed samples and underlying probability distribution is the focus of the concepts which we propose for a bias evaluation. The distinction between these methods is explained in an additional section after presenting the respective approaches.

A. Pearson's Chi-squared Test

In [12], Pearson shows how to test the null hypothesis that observed data fit an assumed reference probability distribution. In the context of HOA-PUFs, we can therefore check how well the observed PUF symbols are consistent with the behaviour of the expected reference PUF.

The χ^2 statistic is computed as

$$\chi^2 = \sum_{i=1}^{N} \frac{(x_i - np_i)^2}{np_i} \tag{1}$$

where n is the sample size and the alphabet of the HOA-PUF contains N symbols. Symbol i has been observed x_i times in the data set and the corresponding reference probability is p_i. It is necessary when computing χ^2 that each symbol is expected to occur at least once given the sample size. The authors state in [13, p. 544] that "most authorities" suggest the expected amount of occurrences to be at least 5 and not less than 3. This means that a sufficient sample size is needed which is especially relevant for HOA-PUFs with equidistant quantization, because some symbols might have a very low probability.

Different to the application of the chi-squared test to PUFs in [2], after computing χ^2, we additionally compute the p-value. The goal is to analyze how likely this result is, given the respective degrees of freedom based on the χ^2 distribution. Degrees of freedom d_f are defined as the amount of variables (in this case N symbols in the alphabet of the HOA-PUF) minus the amount of constraints (here a single one: the fixed sample size n). Hence $d_f = N - 1$. The p-value is determined as a one-sided test utilizing the cumulative probability function of the d_f-dependant χ^2 distribution [13, p. 544]:

$$p = 1 - \mathrm{cdf}_{\chi^2_{d_f}}(\chi^2) \tag{2}$$

If $p < \alpha$ the null hypothesis is rejected, which allows for the conclusion that the data does not fit the expected probability of the reference PUF sufficiently well.

We also suggest a slight modification to achieve a less strict test. There is the possibility to adapt the reference probability (p_i in Equation (1)). Instead of, e.g., a uniform distribution for equiprobable quantization, we want to find a new reference probability distribution with the following two properties: (i) It has to consist of individual p_i such that each p_i is within a user-defined interval $\tilde{p}_l < p_i < \tilde{p}_u$ of acceptable biases[1] and (ii) it has to be the most likely one to result in the observed data. This results in the following optimization problem:

$$\begin{aligned} \max_{\text{all } p_i} L(\mathbf{p}) = \prod_{i=1}^{N} p_i^{x_i} \qquad & \text{subject to} \\ p_i < \tilde{p}_u \qquad & \forall i \in [1; N]\ (I) \\ p_i > \tilde{p}_l \qquad & \forall i \in [1; N]\ (II) \\ \sum_{i=1}^{N} p_i = 1 \qquad & (III) \end{aligned}$$

$L(\mathbf{p})$ is a function of all p_i which computes the probability that the reference probability distribution results in the observed symbol frequencies x_i. N is the order of the PUF response alphabet. (I) and (II) ensure that property (i) is satisfied; (III) ensures that the new reference probability distribution is valid. By maximizing we achieve property (ii). That way, a user can design a less strict Pearson's Chi-squared Test by allowing a slight offset from the original reference distribution.

B. Based on Multinomial Confidence Interval

The second concept and the associated test strategy rely on a *Multinomial Confidence Interval*. For each symbol it states a confidence interval regarding that symbol's probability of occurrence. As a result, we can draw the conclusion that the underlying probability distribution of the PUF under test is contained within the computed intervals with a probability of larger than $1 - \alpha$. Unlike individual binomial confidence intervals, we can therefore address the probability distribution simultaneously with one overall α as confidence level instead of independent ones.

This concept provides several useful aspects for testing: First, we can see if for a given sample size any useful statements about the underlying probability distribution are possible, i.e. if the intervals are sufficiently small; Second, this idea can be reversed; We can also determine the amount of samples needed to achieve confidence intervals of a desired size. Third, we check for each symbol if its expected reference probability is within its corresponding computed confidence interval (cf. Figure 2). If for example, this is not the case for every symbol, this might indicate a low-quality PUF. Fourth, we can use the largest upper bound of all the confidence intervals to compute a lower bound for the min-entropy under the constraint that there is a Type I error probability α.

To compute the multinomial confidence intervals, we follow the approach by Sison and Glaz [14, Theorem 2.1] with slight modifications as suggested in [15]. Sison and Glaz approximate the underlying probabilities with the help of a doubly truncated Poisson distribution. They then iteratively increase the size of all the intervals at once to find the smallest size that still contains the underlying distribution with probability larger than $1 - \alpha$. The authors in [15] propose slightly different bounds to achieve symmetric confidence intervals. They also provide an SAS code as an implementation guideline[2]. While there are also different approaches for multinomial confidence intervals, the method by Sison and Glaz is preferred for PUFs, because they achieve smaller intervals than others [14], [15].

In [14] the authors also address the issue of determining the required sample size for a desired size of the confidence intervals. To compute the required sample n_{req} size given the desired width $2w$ of each confidence interval according to [14], we first start with estimates of the underlying probability distribution $\hat{p}_i$. This could for example either be based on the probability distribution of the reference PUF or utilize the maximum-likelihood estimator (relative frequency of symbols) on a given small sample size. Due to the approximation in [14], we therefore iteratively increase n_{req} in

$$p := Pr\left\{\lceil n_{req}\hat{p}_i - n_{req}w \rceil \leq X_i \leq \lfloor n_{req} \rfloor\right\} \tag{3}$$

until $p \geq 1 - \alpha$.

C. Based on "Acceptable Intervals"

The third approach also involves hypothesis testing. It is similar to the hypothesis test in [7] but extended to HOA-PUFs. The goal is to define so-called *Acceptable Intervals*. If for a data set the number of occurrences for each symbol is within a pre-computed interval, we accept the PUF as a high-quality PUF. The null hypothesis H_0 that we want to reject based on the data samples is that we have a low-quality PUF. If we manage to reject H_0, we accept the alternative hypothesis of having a high-quality PUF. Thus, we conclude that the probability of the underlying PUF being of a high quality is larger than $1 - \alpha$.

The null hypothesis "The PUF is a low-quality PUF" is a composite of multiple different hypotheses. When presuming only bias effects in the PUF, by definition, a low-quality PUF has at least one symbol for which the probability p_i is either too high ($p_i > p_{i,u}$) or too low ($p_i < p_{i,l}$). Thus, given a HOA-PUF with alphabet size N, checking $N \cdot 2$ individual null hypotheses is required. Only rejecting all of them *at once* allows for deducing a high-quality of the PUF.

However, there is a discrepancy between the α of the overall null hypothesis α_{all} and the one for the symbol-wise testing $\alpha_{i,u}$ and $\alpha_{i,l}$. This is due to the problem of multiple testing: The more hypothesis tests are performed on the same set of data, the more likely it is to reject the null hypothesis although it is correct [16]. Correcting the α's of the individual tests mitigates this effect. For this purpose Bonferroni correction is used to get $\alpha_i = \alpha_{i,u} = \alpha_{i,l}$ with

$$\alpha_i = \frac{\alpha_{all}}{N \cdot 2}. \tag{4}$$

[1] This approach allows for testing realistic non-ideal PUFs. However, they then strictly require post-processing to deal with the PUF weaknesses accepted by the test. This is because the finally derived secret should be perfectly random, whereas a realistic PUF response most likely is non-ideal.

[2] We have found a typo in the code in [15]: On page 18 directly above the lines computing $vol1$ and $vol2$ it should say "if out[i,4]<0 then out[i,4]=0" instead of "if out[i,2]<0 then out[i,2]=0 " and "if out[i,5]>1 then out[i,5]=1" instead of "if out[i,3]>1 out[i,3]=1".

The individual hypotheses for each symbol are now analyzed with the corrected α_i as presented for binary PUFs in [7]. We hereby exploit the fact that each symbol of the HOA-PUF can also be tested in a binomial setting: each response symbol either is or is not the symbol under test and either does or does not appear in a sample. Thus, the p-values of the hypothesis tests for each symbol are:

$$p = Pr\{X_i \leq x_{i,u}\} = \sum_{j=0}^{x_{u,i}} \binom{n}{j} p_{i,u}^{j} (1-p_{i,u})^{n-j} \quad (5)$$

$$p = Pr\{X_i \geq x_{i,l}\} = \sum_{j=x_{i,l}}^{n} \binom{n}{j} p_{i,l}^{j} (1-p_{i,l})^{n-j} \quad (6)$$

Hereby, n denotes the sample size, i the index of the symbol under test, X_i the random variable for the frequency of symbol i in the data set. $[p_{i,l}; p_{i,u}]$ is an interval, in which the underlying probability p_i would be considered acceptable. Ensuring that the p-values remain less than α_i allows for computing the smallest and largest acceptable occurrence of a symbol $x_{i,u}$ and $x_{i,l}$. These variables finally define the so-called Acceptable Interval $[x_{i,l}; x_{i,u}]$ for symbol i. A frequency of symbol i between $x_{i,l}$ and $x_{i,u}$ times corresponds to an underlying p_i within the predefined acceptable range with probability of $> 1-\alpha$. Note that unlike the binary case for which the one-probability should be 0.5, the computed acceptable interval is not symmetric around the expected reference value.

In this test, there are three important parameters for the evaluation: α for the whole hypothesis test, the sample size n, and the respective $p_{i,u}$ and $p_{i,l}$. Not every combination of these parameters is valid. For example, a sample size might be too small such that an Acceptable Interval of a desired small size cannot be achieved. In this case, a user has to increase any of the values, i.e. either provide more samples, accept a larger interval for the underlying probability p_i or tolerate a less strict α. For a larger α the error probability of Type I increases.

D. Differences Between the Proposed Methods

Pearson's Chi-squared Test is less complex to compute than the others. However, it does not provide any insights as to why a HOA-PUF possibly fails this test. Given the resulting p-value, we can come to two conclusions: If we reject the null hypothesis "The data fit the expected probability distribution well", this means that PUF under test probably is of a low quality. If we accept the null hypothesis, no clear statement with regard to the quality is possible. This is because we do not control the type II error probability that we consider a low-quality PUF to be of a high quality.

Unlike the other two approaches, the Multinomial Confidence Interval states that the underlying probability distribution is within the respective interval with at least with probability $1-\alpha$. It does not provide information about the complementary event and it does not involve the concept of a low-quality PUF.

The Acceptable Intervals in Section III-C test the null hypothesis of having a low-quality PUF as opposed to Pearson Chi-squared test assuming a high-quality PUF. By rejecting this null hypothesis, we can therefore infer a high PUF quality.

IV. Application to Real Data

To illustrate the concepts, this section applies the tests to exemplary data based on practical measurements. The parameter choices such as α for a hypothesis test or confidence intervals should be adapted according to the respective use-case of the PUF. However, a high sample size is a prerequisite for a high confidence. In this section, the parameters only serve as an example and the main focus is on applying the methods and understanding the corresponding results. We therefore also compute state-of-the-art metrics which are available for HOA-PUFs to demonstrate that the new approaches provide deeper insights into a PUF's behaviour.

A. Evaluating a HOA-PUF Based on Real Data

For real-world data, we focus on counter differences derived from our own implementation [17] of the Loop PUFs in [18]. While a Loop PUF originally uses the sign of the difference of two oscillation frequency counters and thus targets binary responses, for the sake of this paper, we use equiprobable quantization to provide respective response symbols from a higher-order alphabet as output[3]. The data set contains measurements from 180 BASYS 3 FPGA boards with 48 PUF instances each. The counter differences are for one fixed challenge to exclude any correlation effects between similar challenges and are averaged over 201 repeated measurements to mitigate noise effects. Ideally, this distribution should asymptotically follow a normal distribution (with mean 0). Any offset of the mean in real data can be corrected by shifting the quantization intervals as well. This is done in this work and based on [2, Equation 4].

For a suitable equiprobable quantization, we first determine the underlying normal distribution, which will serve as reference probability distribution $\mathcal{N}_{ref}$ as well. Note that distribution $\mathcal{N}_{ref}$ does not match the data perfectly. Our methods try to detect and evaluate the resulting difference between the reference probability distribution and observed symbol frequencies.

The mean and standard deviation based on the data $x_1, \ldots, x_n$ are

$$\mu_{\mathcal{N}_{ref}} = \frac{\sum_{i=1}^{n} x_i}{n} \text{ and } \sigma_{\mathcal{N}_{ref}} = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \mu_{\mathcal{N}_{ref}})^2}{n-1}}. \quad (7)$$

We define intervals for the counter differences (here assuming an alphabet size of 16) such that according to $\mathcal{N}_{ref}$ each interval has the same probability. Given the intervals, we map each of the measured counter differences to a corresponding symbol of a HOA-PUF.

In Figure 1, the blue bars visualize the frequency of each of the 16 symbols after an equiprobable quantization. The width corresponds to the underlying interval used for quantization. Given the exemplary data, $\mu_{\mathcal{N}_{ref}} \approx -30$ and $\sigma_{\mathcal{N}ref} \approx 212$. In comparison to the red line at around 540 visualizing the expected symbol frequencies, the actual symbol frequencies

[3]This is by no means a secure implementation; it is expected to be susceptible to a Side Channel Attack and is for illustration purposes only.

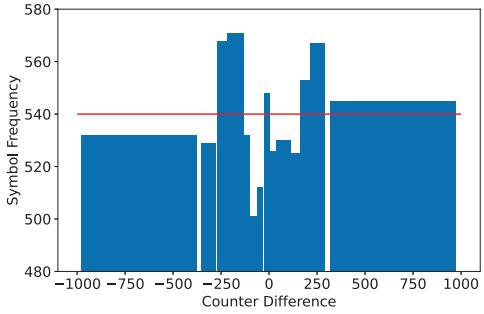

Fig. 1. Histogram of symbol frequency of Loop PUF data after equiprobable quantization over counter differences. The red horizontal line at 540 shows the expected frequency for an ideal uniform distribution.

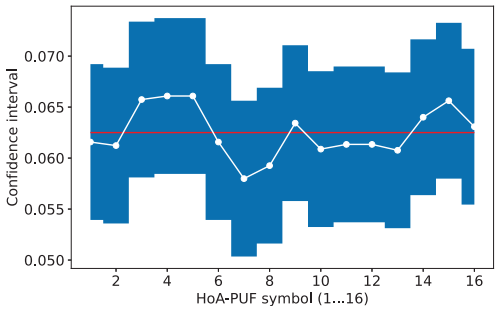

Fig. 2. Multinomial Confidence Interval plotted for the measured data (cf. Figure 1) according to [14]. The red line indicates the expected probability of 0.0625 for a uniform distribution. The dots in white show the empirical probability x_i/n.

differ. E.g., symbol 1 corresponding to any counter difference being less than -355 (for the plot the lower bound is set to -1000) occurs 532 times. The following evaluation checks if the discrepancy between the observed and the expected outcome is critical by applying the suggested methods.

a) Pearson's Chi-Squared Test: First, we apply Pearson's Chi-squared Test to the data set in comparison to the reference probability distribution, i.e. in this analysis a uniform distribution. The null hypothesis H_0 is that the data was indeed drawn from a uniform distribution. We set the significance level $\alpha = 0.05$. This is a value which has to be chosen according to the respective use case of the PUF.

The resulting p-value for the given data is around 0.62. We therefore cannot reject the null hypothesis and thus cannot conclude that the PUF's underlying probability distribution differs from the reference uniform distribution. Because we do not reject the null hypothesis, we do not need to utilize the less strict reference probability distribution. Overall, we can make the following statement: assuming a true null hypothesis, there is a 0.62 probability that the resulting symbols have been drawn from such a uniform distribution. This shows the limitation of Pearson's Chi-squared Test in the PUF context. The test can only provide a strong statement if the null hypothesis is rejected.

b) Multinomial Confidence Interval: To compute the Multinomial Confidence Interval as proposed in [14], we have to input the symbol frequencies as well as a significance level α. This α does not have to be the same α as for the other tests. The resulting confidence interval (for $\alpha = 0.05$) with regard to the actual probability distribution is shown in Figure 2.

So for, e.g., symbol 7 which occurs the least often, the confidence interval still contains the reference probability for a perfect uniform distribution. Because the expected reference probability for a symbol is an element of each confidence interval, this is another indication for a high-quality PUF. Any quantitative metric only utilizing a single empirically determined probability $\hat{p} = \frac{x_i}{n}$ value for each symbol cannot provide the same confidence in the overall result.

We can use the largest probability in the Multinomial Confidence Interval to estimate the min-entropy. For symbol 4 the upper bound of the interval is 0.073 and we have

$$H_\infty(X) = -\log \max_i p_i = -\log p_4 \approx 3.76(bit) \qquad (8)$$

which is a lower bound with probability $1 - \alpha = 0.95$ given the underlying confidence level.

On a mere technical side, note that all intervals have same size because all symbols are considered at once in a multinomial approach in contrast to a binomial one. In comparison to other approaches, [14] achieve a smaller volume of the confidence region which is the product of all interval sizes. If instead of $(0.0152)^{16}$ a smaller volume (and thus smaller confidence intervals) are desired, more samples would be required according to the sample size determination proposed in [14]. E.g., confidence intervals of size 0.01 would require at least 20360 samples.

c) Acceptable Intervals: To compute the Acceptable Intervals, we have to define α_{all}, in the following $\alpha_{all} = 0.01$.[4] We also configure p_l and p_u so that a HOA-PUF symbol fails if it is not likely to have actual probability p_i within the interval ± 0.0125 of the reference $1/16 = 0.0625$.

As shown in Figure 3 symbols $3, 4, 5, 7, 15$ all are not within their respective green Acceptable Interval. For all other symbols, we have rejected the null hypotheses that $p_i < p_{l,i}$ and $p_i > p_{u,i}$. If all null hypotheses were rejected, it would have been possible to state that the overall HOA-PUF has probabilities in the predefined intervals at least with probability $1 - \alpha_{all}$. One option to further proceed is to repeat the test with less strict parameters; replacing, e.g. interval ± 0.0125 by ± 0.015 results in all passes. This would allow for more fine-grained labeling of the PUF, e.g., as a PUF of medium quality failing the stricter test but passing the less strict one.

The main difference to the other two methods is the null hypothesis. Unlike before, the null hypothesis for the Acceptable Intervals assumes a low quality. If we reject this hypothesis, we can deduce a high quality.

[4]The α for the significance or confidence level of the proposed methods does not have to be the same. Additionally due to multiple testing, an increased α hardly has any impact regarding the Acceptable Intervals. The individual α_i are small anyway.

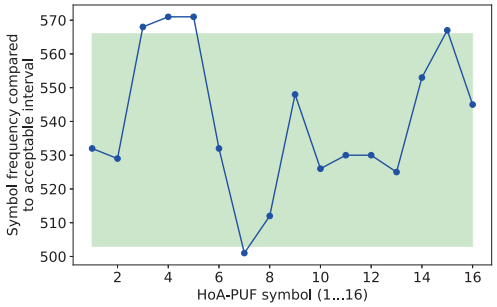

Fig. 3. So-called Acceptable Intervals in green as well as the symbol frequencies of the HoA-PUF.

Independent of the data set, this exemplary analysis shows the different flavors of the tests and that different kinds of complementary statements are possible when applying all of them. Overall, given the exemplary data set and test parameter choices it is visible for the three approaches that the Pearson's Chi-square Test and the Multinomial Confidence Interval do not indicate a weakness in the HoA-PUF regarding the probabilities of the individual symbols.

B. Comparison to state-of-the-art HoA-PUF metrics

In [9] a modification to the Uniqueness by [4] is suggested. Hereby, the Hamming distance is understood not for binary responses but also for responses of HoA-PUFs. A normalization ensures that the ideal value is 50%. The normalized higher-order Uniqueness for a uniform distribution of the higher-order symbols is computed as

$$u = \frac{N}{(N-1)k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{d_{Hamming}(r_i^l, r_j^l)}{l} \cdot 100\%. \tag{9}$$

r_i^l denotes the PUF response on device i of length l. For the exemplary data set $u = 0.500$ which is the ideal value. Applying the Uniqueness therefore indicates a high-quality PUF. However, in comparison to the proposed metrics, no more fine-grained statements are possible based on this single value. Overall, the proposed metrics also evaluate the PUF to be of rather high-quality, but e.g. the Acceptable Intervals also show that there still is a gap between the actual PUF and a perfectly uniform probability distribution. Please note that a Uniqueness different from 50% does not indicate why this is the case.

An alternative to higher-order Uniqueness is to compute $\chi^2 = 12.7$ as in [2]. However, this value provides only few insights about the bias of a HoA-PUF. Statistically sound conclusions are only possible with the presented extensions.

V. Conclusion

In this work, we have presented three methods to analyze HoA-PUFs with regard to their bias. Unlike many metrics in literature, the novel schemes provide a confidence attached to the results. The metrics cover complementary aspects regarding the statistical bias properties of a HoA-PUF so that we suggest to use them. Our results demonstrate the significance of the new metrics and how they are applied. In the light of emerging applications in the PUF domain, the suggested metrics provide a means to fair comparison and quality assurance of new PUF technologies.

Acknowledgement

This work was partly funded by the Bavarian Ministry of Economic Affairs, Regional Development and Energy as part of the project 6G Future Lab Bavaria.

References

[1] M. Majzoobi, F. Koushanfar, and M. Potkonjak, "Lightweight secure pufs," in *2008 IEEE/ACM International Conference on Computer-Aided Design*, 2008, pp. 670–673.

[2] H. Mandry, A. Herkle, S. Müelich, J. Becker, R. F. Fischer, and M. Ortmanns, "Normalization and multi-valued symbol extraction from ro-pufs for enhanced uniform probability distributions," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 12, pp. 3372–3376, 2020.

[3] V. Immler, J. Obermaier, K. K. Ng, F. X. Ke, J. Lee, Y. P. Lim, W. K. Oh, K. H. Wee, and G. Sigl, "Secure physical enclosures from covers with tamper-resistance," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2019, no. 1, p. 51–96, Nov. 2018.

[4] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, "A large scale characterization of ro-puf," in *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*. IEEE, 2010, pp. 94–99.

[5] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh, "Quantitative and statistical performance evaluation of arbiter physical unclonable functions on fpgas," in *2010 International conference on reconfigurable computing and FPGAs*. IEEE, 2010, pp. 298–303.

[6] "Information security, cybersecurity and privacy protection - Physically unclonable functions - Part 2: Test and evaluation methods," ISO / IEC, Standard, Mar. 2021.

[7] F. Wilde and M. Pehl, "On the confidence in bit-alias measurement of physical unclonable functions," in *2019 17th IEEE International New Circuits and Systems Conference (NEWCAS)*. IEEE, 2019, pp. 1–4.

[8] F. Wilde, B. M. Gammel, and M. Pehl, "Spatial correlation analysis on physical unclonable functions," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 6, pp. 1468–1480, 2018.

[9] V. C. Immler, "Higher-order alphabet physical unclonable functions," Ph.D. dissertation, Technische Universität München, 2019.

[10] M. Pehl, T. Tretschok, D. Becker, and V. Immler, "Spatial context tree weighting for physical unclonable functions," in *2020 European Conference on Circuit Theory and Design (ECCTD)*, 2020, pp. 1–4.

[11] V. Immler and K. Uppund, "New insights to key derivation for tamper-evident physical unclonable functions," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2019, no. 3, p. 30–65, May 2019.

[12] K. Pearson, "X. on the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 50, no. 302, pp. 157–175, 1900.

[13] R. M. Heiberger and B. H. Burt Holland, *Statistical Analysis and Data Display An Intermediate Course with Examples in R*. Springer, 2015.

[14] C. P. Sison and J. Glaz, "Simultaneous confidence intervals and sample size determination for multinomial proportions," *Journal of the American Statistical Association*, vol. 90, no. 429, pp. 366–369, 1995.

[15] W. L. May, W. D. Johnson *et al.*, "Constructing two-sided simultaneous confidence intervals for multinomial proportions for small counts in a large number of cells," *Journal of Statistical Software*, vol. 5, no. 6, pp. 1–24, 2000.

[16] H. Abdi, "The Bonferonni and Sĭdák corrections for multiple comparisons. encycl meas stat. 2007; 1: 1–9."

[17] L. Tebelmann, J.-L. Danger, and M. Pehl, "Self-secured puf: protecting the loop puf by masking," in *International Workshop on Constructive Side-Channel Analysis and Secure Design*. Springer, 2020, pp. 293–314.

[18] Z. Cherif, J.-L. Danger, S. Guilley, and L. Bossuet, "An easy-to-design PUF based on a single oscillator: the loop PUF," in *2012 15th Euromicro Conference on Digital System Design*. IEEE, 2012, pp. 156–162.